

Yet Another Trading Algorithm

Technical Manual

Grandez

2021-05-21

Contents

1	Prerequisites	5
	Introduccion	7
2	Technical Decissions	9
2.1	Componentes Web (<i>widgets</i>)	9
2.2	About	10
2.3	Bloque de decisiones una	11
2.4	YATADB	11

Chapter 1

Prerequisites

```
install.packages("bookdown")  
library(abind)  
library(png)  
library(grid)  
library(fontawesome)  
  
# or the development version  
# devtools::install_github("rstudio/bookdown")
```


Introduccion

XXX Aqui la introduccion

Chapter 2

Technical Decissions

2.1 Componentes Web (*widgets*)

Como ya se ha indicado, las páginas Web están basadas en Shiny el cual aparte de definir de manera unívoca el método de programación (basado en los conceptos *reactive* y *bindings*) ofrece un conjunto de *widgets* ya diseñados y listos para usar.

Sobre Shiny se utilizan diferentes paquetes que lo enriquecen de alguna u otra manera, evitando tener que diseñar widgets que ya han sido diseñados y probados, con la limitación de tener que adscribirse a su nomenclatura e implementación.

Normalmente estos paquetes o bien sobrecargan las funciones nativas de Shiny o, lo mas común, utilizan algun criterio de nomenclatura que no colisione con Shiny; por ejemplo, Shiny ofrece una función “alert” y el paquete “*shinywidgets*” ofrece la misma funcionalidad con una versión con mas opciones que denomina “shinyAlert”.

2.1.1 Caso de uso

A lo largo del ciclo de vida del sistema es posible que alguno de los paquetes utilizados dejen de existir o de estar soportados, o que se liberen otros paquetes con otras funcionalidades o mejoras.

Por ejemplo, supongamos:

Primero; que disponemos de un paquete que implementa un UI mas atractivo para los botones que el actual, el nuevo paquete denomina a los botones `prettyButtonxxx` y los actuales se denominan `simpleButton`

Segundo; que aunque no teníamos las herramientas, por cuestiones de diseño y programación contemplamos diferentes botones, digamos uno para aceptar una

operación y otro para anularla. Como no tenemos las herramientas los dos son iguales, pero de manera proactiva hemos creado dos funciones: `btnOK` y `btnKO`

En el primer caso, para implementar el nuevo paquete deberíamos analizar todo el sistema en busca de las funciones `simpleButton`, cambiarlos por el equivalente `prettyButton` y posiblemente ajustar los parámetros de llamada. Lo cual, según el paquete a integrar podría ser, aparte de arriesgado y costoso, directamente inviable.

Por otro lado, fijémosnos en el segundo caso, en este no estamos invocando directamente a `simpleButton` si no a una función propia que actualmente son iguales, por lo que para cambiar el comportamiento de, digamos `btnKO` solo deberíamos modificar una función y estaría disponible para todo el sistema.

2.1.2 Decisión

El enfoque utilizado para solventar este problema es el descrito en el ejemplo anterior.

En lugar de invocar directamente a los widgets de Shiny o del resto de paquetes, **todos y cada uno de ellos** están encapsulados en funciones propias del sistema, de manera que la dependencia de este con los paquetes solo existen en un módulo determinado.

El criterio utilizado es el siguiente:

- Los componentes de diseño se denominan `yuiCamelCaseName`
- Los componentes de servidor se denominan `updCamelCaseName`

Donde, la función `yui` encapsula el widget asociado y la función `upd` gestiona la interacción con ese componente.

De manera habitual, pero no en todos los casos, el nombre del par de funciones se corresponderá con su correspondiente widget o de la funcionalidad que implementa; por ejemplo, si tenemos un boton generico, se llamará `yuiButton` aunque podemos tener otros mas botones mas especificos como `yuiButtonOK` o `yuiButtonKO`

2.1.3 Wrappers

qa1

2.2 About

En esta sección se detallan algunas de las decisiones de diseño, técnicas, etc. que se han tomado, así como el porqué y sus alternativas.

El objetivo es mantener la trazabilidad y ofrecer un mejor conocimiento del sistema, evitando “repensar” de manera repetida la solución a ciertos problemas.

Por facilidad, cada decisión va en su propio archivo

2.3 Bloque de decisiones una

Aqui se indican las decisiones que se han tomado y por que

2.4 YATADB

This is the package encapsulating the persistence layer